LLVM + MLIR + CIRCT: A golden age of HLS?

Masanori Ogino

Online HLS Workshop on 2021-09-22

Hiroshima University, Experimental Quark Physics Laboratory, Computing Team

Table of Contents

1. The Age of Heterogeneous Computing
2. LLVM: An Intermediate Representation for CPU
3. MLIR: A Framework for Intermediate Representations
4. CIRCT: A Circuit Design Toolkit on Top of MLIR
5. Idea: A Type-safe Language for HLS?
6. Conclusion

Heterogeneous Computing

Is Free Lunch Finally Over?

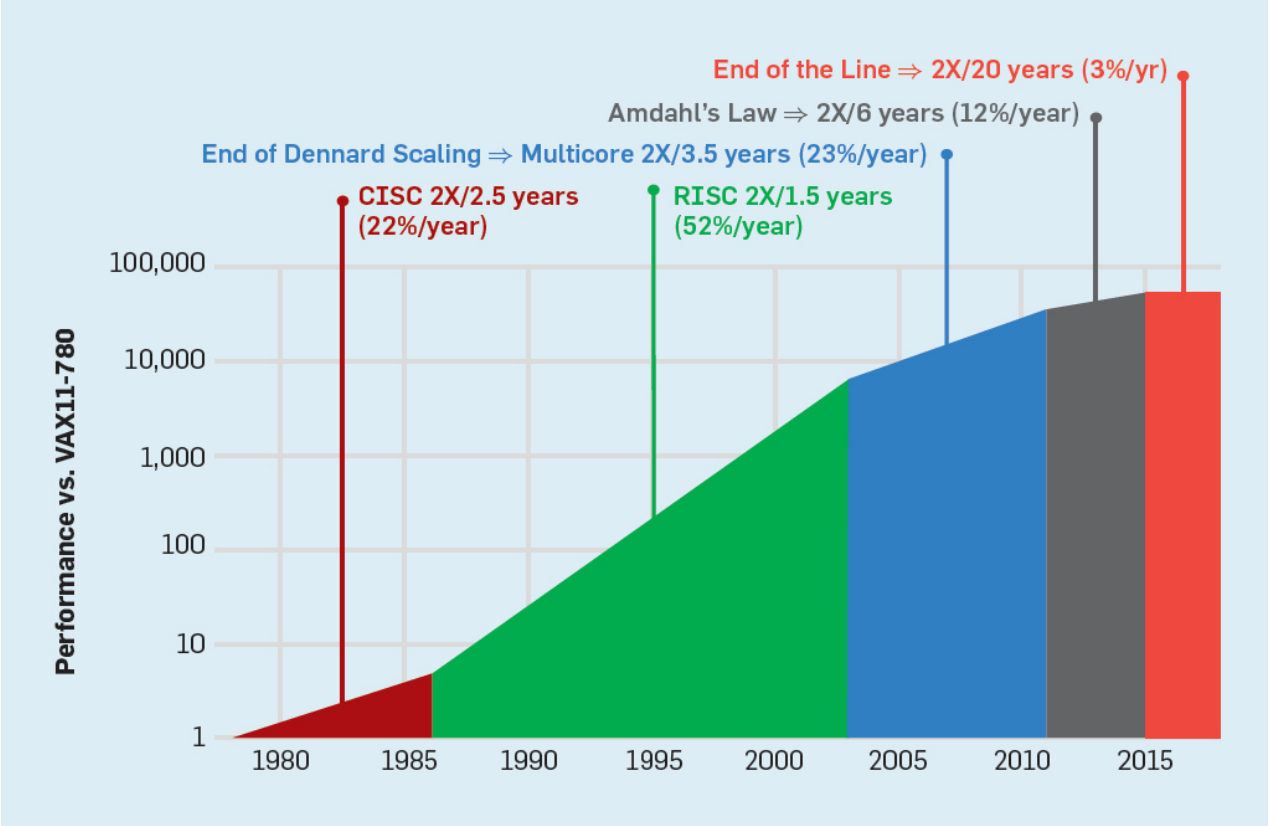


Figure 1: Growth of computer performance using integer programs (SPECintCPU), cited from J. L. Hennessy and D. A. Patterson, "A New Golden Age for Computer Architecture", CACM Feb. 2019.

The Age of Heterogeneous Computing

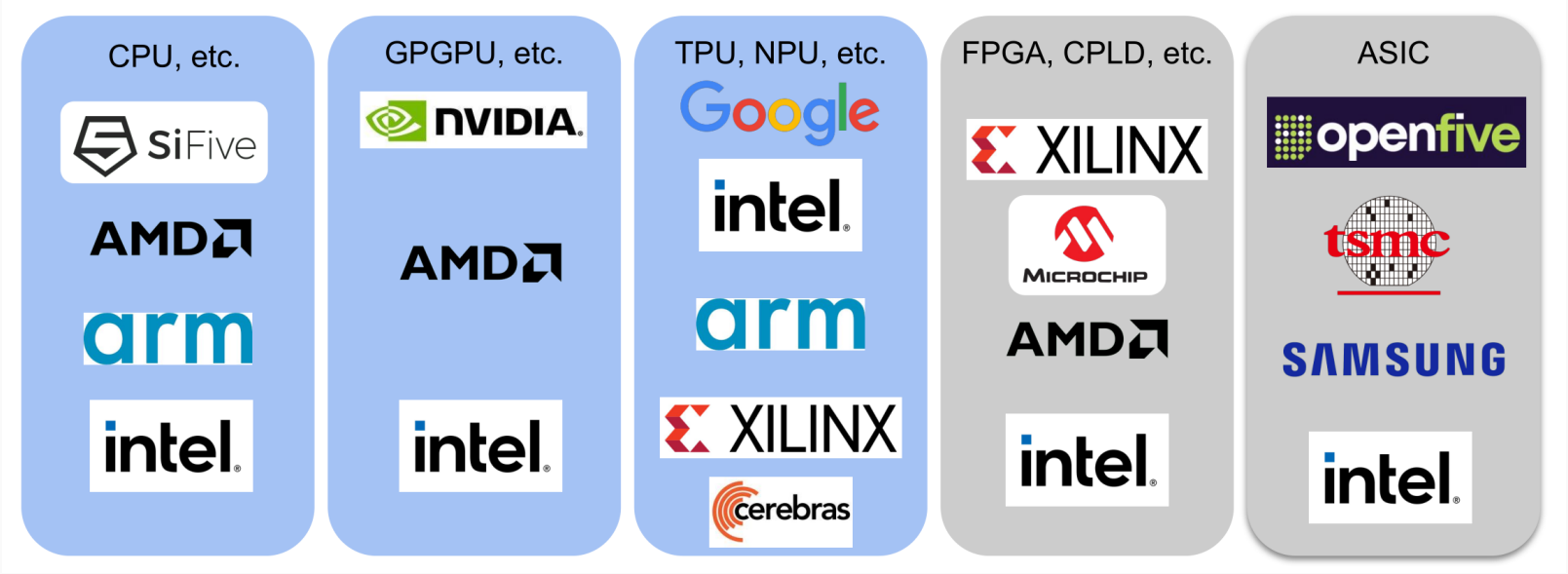


Figure 2: Major players on heterogeneous computing market, cited from C. Lattner, "The Golden Age of Compilers in an era of Hardware/Software co-design", ASPLOS 2021.

A Big Problem: Fragmentation of Tools, Languages, ...

Figure 3: Tools around heterogeneous computing, cited from C. Lattner, "The Golden Age of Compilers in an era of Hardware/Software co-design", ASPLOS 2021.

LLVM

What Is LLVM?

What Is LLVM?

- A **modular** compiler toolkit
- **Intermediate representation** (IR) for **abstract CPU**
- Can use for **SIMT GPGPU**
- Not good for **massively parallel accelerators** (TPU, NPU, ...)

MLIR

What is MLIR?

What Is MLIR?

- A toolkit for **designing IRs**
- **Conversion** between different **abstraction levels**
- Define **IRs for GPU, TPU, NPU, … by ourselves**
- Even LLVM IR could be defined on top of MLIR

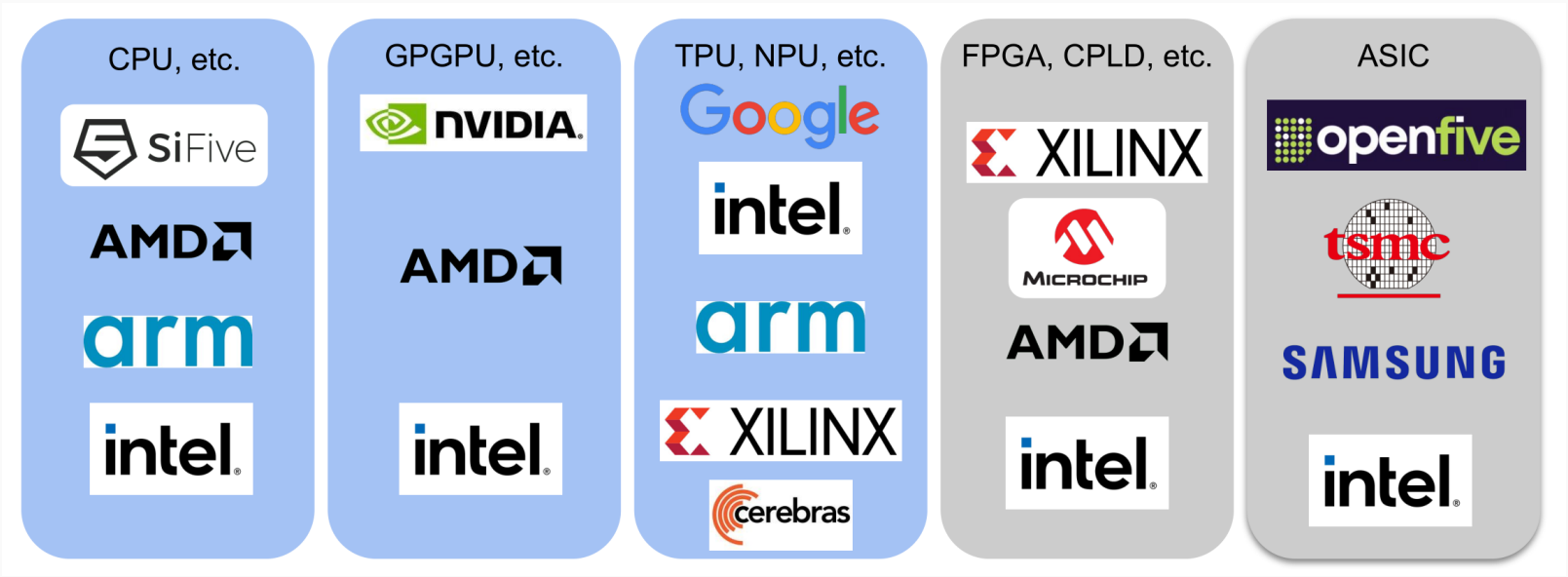
CIRCT

What Is CIRCT?

What Is MLIR?

- IRs for **describing digital circuits** on top of MLIR
- Affine, sequential, combinational, …

A Foundation of Heterogeneous Computing



LLVM + MLIR + CIRCT

A foundation of next-generation programming environment?

HLS

HLS: A Territory of C

Table 1: Overview of high-level synthesis tools, cited from R. Nane et al., "A Survey and Evaluation of FPGA High-Level Synthesis Tools", TCAD 2015.

Status	Compiler	Owner	License	Input	Output	Year	Domain	TestBench	FP	FixP
In Use	eXCite	Y Explorations	Commercial	C	VHDL/Verilog	2001	All	Yes	No	Yes
	CoDeve-loper	Impulse Accelerated	Commercial	Impulse-C	VHDL Verilog	2003	Image Streaming	Yes	Yes	No
	Catapult-C	Calypto Design Systems	Commercial	C/C++ SystemC	VHDL/Verilog SystemC	2004	All	Yes	No	Yes
	Cynthesizer	FORTE	Commercial	SystemC	Verilog	2004	All	Yes	Yes	Yes
	Bluespec	BlueSpec Inc.	Commercial	BSV	SystemVerilog	2007	All	No	No	No
	CHC	Altium	Commercial	C subset	VHDL/Verilog	2008	All	No	Yes	Yes
	CtoS	Cadence	Commercial	SystemC TLM/C++	Verilog SystemC	2008	All	Only cycle accurate	No	Yes
	DK Design Suite	Mentor Graphics	Commercial	Handel-C	VHDL Verilog	2009	Streaming	No	No	Yes
	GAUT	U. Bretagne	Academic	C/C++	VHDL	2010	DSP	Yes	No	Yes
	MaxCompiler	Maxeler	Commercial	MaxJ	RTL	2010	DataFlow	No	Yes	No
	ROCCC	Jacquard Comp.	Commercial	C subset	VHDL	2010	Streaming	No	Yes	No
	Synphony C	Synopsys	Commercial	C/C++	VHDL/Verilog SystemC	2010	All	Yes	No	Yes
	Cyber-WorkBench	NEC	Commercial	BDL	VHDL Verilog	2011	All	Cycle/ Formal	Yes	Yes
	LegUp	U. Toronto	Academic	C	Verilog	2011	All	Yes	Yes	No
	Bambu	PoliMi	Academic	C	Verilog	2012	All	Yes	Yes	No
	DWARV	TU. Delft	Academic	C subset	VHDL	2012	All	Yes	Yes	Yes
	VivadoHLS	Xilinx	Commercial	C/C++ SystemC	VHDL/Verilog SystemC	2013	All	Yes	Yes	Yes
N/A	Trident	Los Alamos NL	Academic	C subset	VHDL	2007	Scientific	No	Yes	No
	CHiMPS	U. Washington	Academic	C	VHDL	2008	All	No	No	No
	Kiwi	U. Cambridge	Academic	C#	Verilog	2008	.NET	No	No	No
	gcc2verilog [45]	U. Korea	Academic	C	Verilog	2011	All	No	No	No
	HercuLeS	Ajax Compiler	Commercial	C/NAC	VHDL	2012	All	Yes	Yes	Yes
Abandoned	Napa-C	Sarnoff Corp.	Academic	C subset	VHDL/Verilog	1998	Loop	No	No	No
	DEFACTO	U. South Cailf.	Academic	C	RTL	1999	DSE	No	No	No
	Garp	U. Berkeley	Academic	C subset	bitstream	2000	Loop	No	No	No
	MATCH	U. Northwest	Academic	MATLAB	VHDL	2000	Image	No	No	No
	PipeRench	U.Carnegie M.	Academic	DIL	bitstream	2000	Stream	No	No	No
	SeaCucumber	U. Brigham Y.	Academic	Java	EDIF	2002	All	No	Yes	Yes
	SA-C	U. Colorado	Academic	SA-C	VHDL	2003	Image	No	No	No
	SPARK	U. Cal. Irvine	Academic	C	VHDL	2003	Control	No	No	No
	AccelDSP	Xilinx	Commercial	MATLAB	VHDL/Verilog	2006	DSP	Yes	Yes	Yes
	C2H	Altera	Commercial	C	VHDL/Verilog	2006	All	No	No	No
	CtoVerilog	U. Haifa	Academic	C	Verilog	2008	All	No	No	No

Idea: Turning LLVM-based programming language(s) into HLS language(s)?

- LLVM-based compiler is everywhere
- C, C++, Fortran, **Rust**, Swift, **Julia**, **Common Lisp**, **Haskell**, **TensorFlow**, ...
- Use those powerful languages into HLS with MLIR and CIRCT?

Conclusion

Conclusion

- Fragmentation of tools and languages
- Re-implementation and reinvent again and again
- LLVM + MLIR + CIRCT: a breakthrough?
- HLS with something better than C?

Questions, suggestions, corrections, whatever?

Thank you for attending!